SWEN90016

Software Processes & Project Management

Quality Assurance
Planning, Control, Monitoring

2019 – Semester 1

Tutorial 8

Today's aim

1. Know quality compliance frameworks, and their pros and cons

2. Know the different tools for achieving high quality outcomes

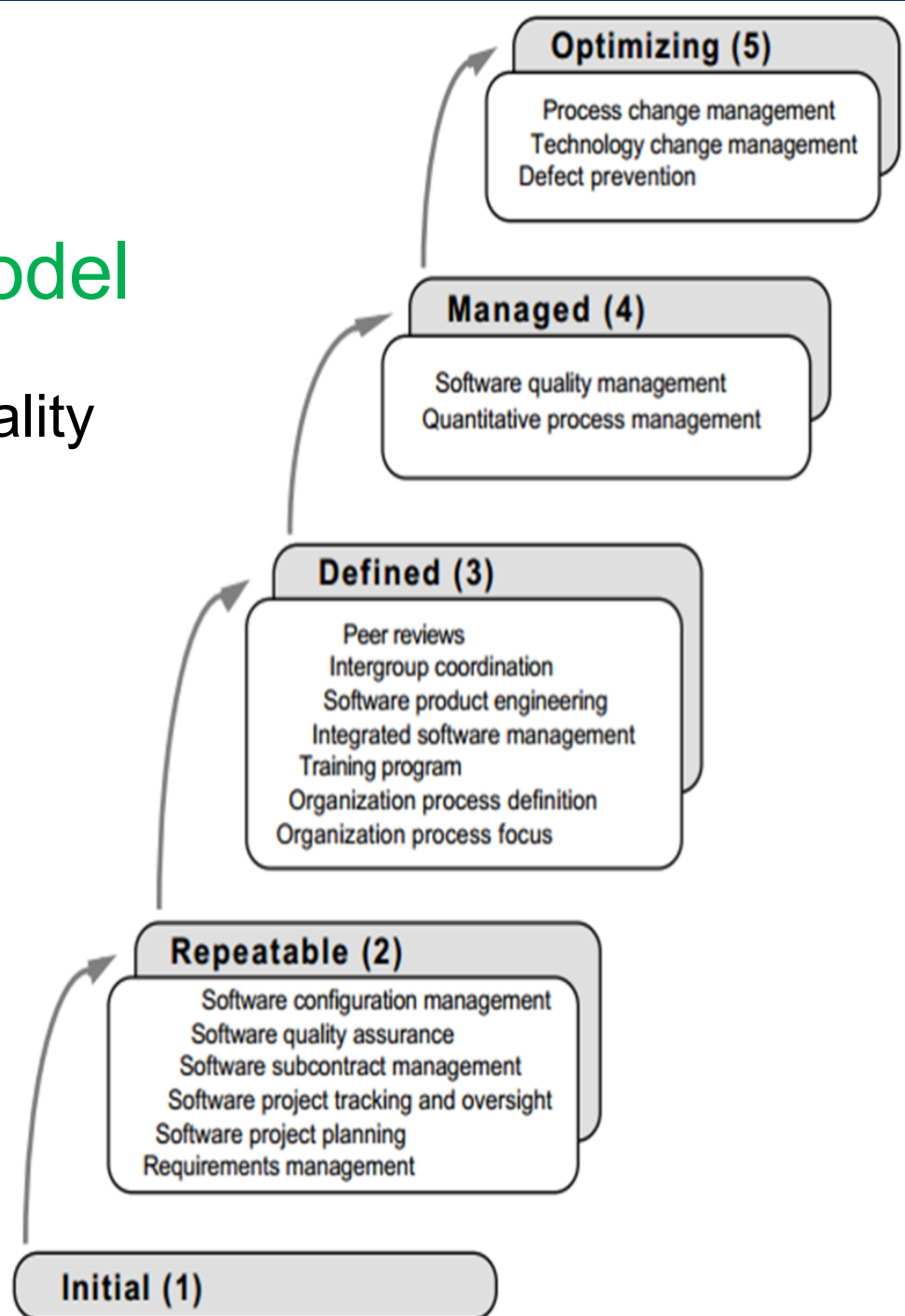3. Discuss the costs and benefits of quality

1. What are the advantages of moving a software process from "repeatable" to "defined" ? (choose all that are correct)

a) A defined software process has measurement and control processes in place to encourage good management of the project

b) Less costly on the project team

c) It enables less experienced managers to perform as well as highly experienced managers.

d) Good for team morale

MELBOURNE

2. What are the disadvantages of moving a software process from "repeatable" to "defined" ? (choose all that are correct)

a) It is difficult to manage because there are fewer guidelines

b) Expensive

c) It does not provide opportunities for junior managers to gain skills

d) Can be problematic for team morale and interpersonal relationships
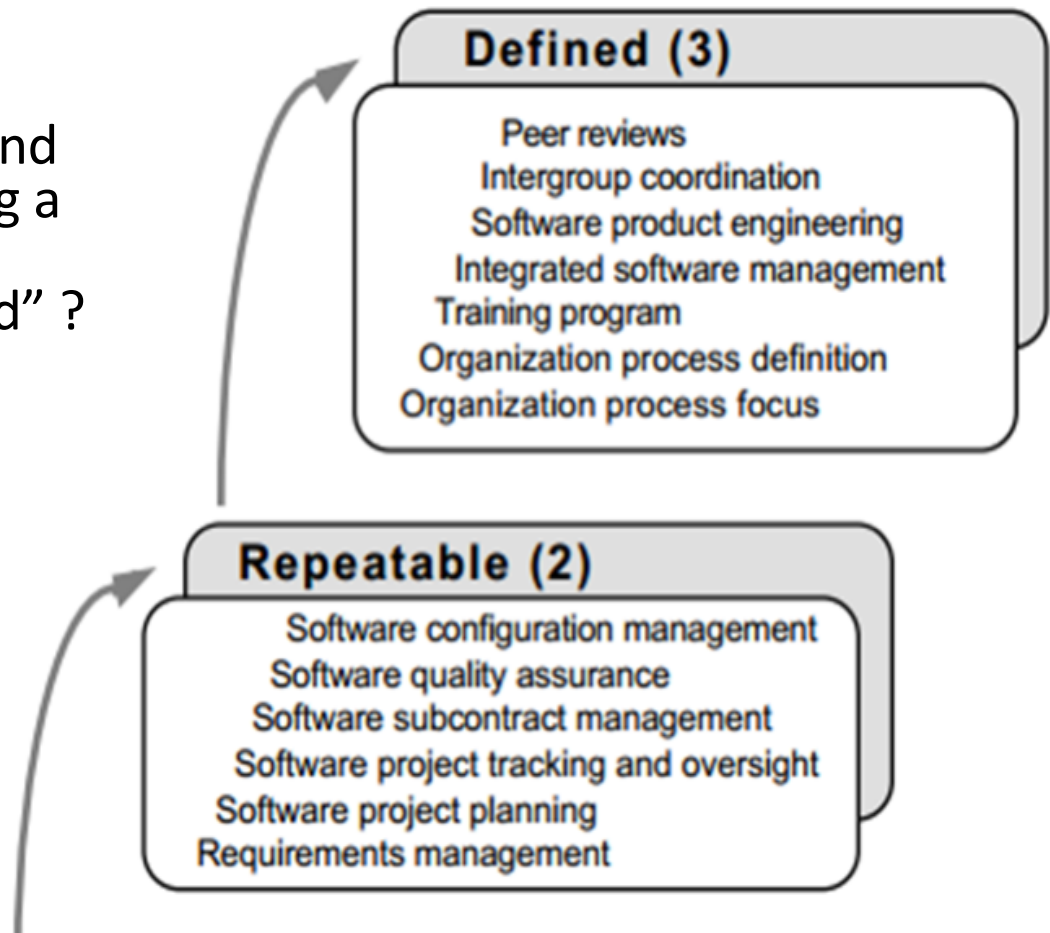
Capability Maturity Model

A tool for achieving high quality

Optimizing (5)
- Process change management
- Technology change management
- Defect prevention

Managed (4)
- Software quality management
- Quantitative process management

Defined (3)
- Peer reviews
- Intergroup coordination
- Software product engineering
- Integrated software management
- Training program
- Organization process definition
- Organization process focus

Repeatable (2)
- Software configuration management
- Software quality assurance
- Software subcontract management
- Software project tracking and oversight
- Software project planning
- Requirements management

Initial (1)

THE UNIVERSITY OF
MELBOURNE

From Lecture 9, slide 35

1. What are the advantages and disadvantages of moving a software process from "repeatable" to "defined" ?

Defined (3)

Peer reviews
Intergroup coordination
Software product engineering
Integrated software management
Training program
Organization process definition
Organization process focus

Repeatable (2)

Software configuration management
Software quality assurance
Software subcontract management
Software project tracking and oversight
Software project planning
Requirements management

In your assignment groups, **discuss** quality processes & do these activities

2. Create an appropriate formal checklist to review your own group assignment

3. Describe the outcome of this review?

THE UNIVERSITY OF
MELBOURNE

A tool

Quality Attributes	Definition According to McCall et al.
Correctness	The extent to which a program satisfies its specifications and fulfils the user's mission objectives.
Reliability	The extent to which a program can be expected to perform its intended function with required precision.
Efficiency	The amount of computing resources and code required by a program to perform a given function.
Integrity	The extent to which access to software or data by unauthorised persons can be controlled.
Usability	The effort required to learn, operate, prepare input, and interpret output of a program.
Maintainability	The effort required to locate and fix an error in an operational program.
Testability	The effort required to test a program to ensure that it performs its intended function.
Flexibility	The effort required to modify an operational program.
Portability	The effort required to transfer a program from hardware and/or software environment to another.
Reusability	The extent to which a program (or parts thereof) can be reused in other applications.
Interoperability	The effort required to couple one system with another.

A tool

Checklist for software requirements specification artifact

Organisation and Completeness

☐ Are all internal cross-references to other requirements correct?

☐ Are all requirements written at a consistent and appropriate level of detail?

☐ Do the requirements provide an adequate basis for design?

☐ Is the implementation priority of each requirement included?

☐ Are all external hardware, software, and communication interfaces defined?

☐ Have algorithms intrinsic to the functional requirements been defined?

☐ Does the specification include all of the known customer or system needs?

☐ Is the expected behaviour documented for all anticipated error conditions?

Correctness

☐ Do any requirements conflict with or duplicate other requirements?

☐ Is each requirement written in clear, concise, unambiguous language?

☐ Is each requirement verifiable by testing, demonstration, review, or analysis?

☐ Is each requirement in scope for the project?

☐ Is each requirement free from content and grammatical errors?

☐ Is any necessary information missing from a requirement? If so, is it identified as "to be decided"?

☐ Can all of the requirements be implemented within known constraints?

☐ Are any specified error messages unique and meaningful?

Quality Attributes

☐ Are all performance objectives properly specified?

☐ Are all security and safety considerations properly specified?

☐ Are other pertinent quality attribute goals explicitly documented and quantified, with the acceptable tradeoffs specified?

Traceability

☐ Is each requirement uniquely and correctly identified?

☐ Is each software functional requirement traceable to a higher-level requirement (e.g., system require-

Quick video when on the importance of SW quality

https://www.youtube.com/watch?v=x8JR9Lh4XJ0

What is the role of QA in agile?

a) After development there is a separate testing done by the agile team in a number of sprints

a) As there is fast development cycles there is no time for testing

b) Agile aims to adapt to changes quickly and minimize time so there is no testing

a) Testing is done in each sprint

e) Continuous integration between development and testing

Agile QA desk-audit hurdle:

Invite multi skilled audience to desk audit: a business analyst, another developer and a tester

Review the code at the developer's desk, before the code is allowed to be committed into the shared *git* repository, GITHub.

Once the code is committed into GITHub, it's test suite is run immediately by the Continuous Integration tool

CI tool displays run code's pass/fail status

Sprint Review QA evaluation:

What
Where
Who
When
Why

• Build small piece of working software with minimal features

• Showcase the product chunk to the stakeholders **early**

• Fail **fast** and as cheaply as possible, & get timely feedback

• Capture the ***technical debt item*** in the Product Backlog, (optionally in FDD format)

• The Product Owner sets the priority of the ***technical debt item***

Write QA Requirements as User Stories

As the Agile Scrum team:

5 a. *We want a Quality Plan, so that our Sprint has a strong Quality Management focus*

As a Quality Assurance **Design** team:

b. *We want a QA checklist, so that key categories and attributes are assessed at defined times*

As the System Administrator:

c. *I want a password policy guideline, so that our application has helpful processes*

I want a password policy checklist, so that our application is highly secure

Done!

ISO-9126:
SW Quality

A compliance framework for achieving high quality

http://www.sqa.net/iso9126.html

Characteristics	Sub-characteristics
Functionality	Suitability
	Accuracy
	Interoperability
	Security
Reliability	Maturity
	Fault Tolerance
	Recoverability
Usability	Understandability
	Learnability
	Operability
	Attractiveness
Efficiency	Time Behaviour
	Resource Utilisation
Maintainability	Analysability
	Changeability
	Stability
	Testability
Portability	Adaptability
	Installability
	Co-existence
	Replaceability